JURNAL ILMIAH INFORMATIKA GLOBAL VOLUME 7 No.1 JULI 2016
ISSN PRINT : 2302-500X
ISSN ONLINE : 2477-3786

RANCANG BANGUNG APLIKASI ENKRIPSI DENGAN MENGGUNAKAN METODE RSA BERBASIS *WEB*

Ricky Maulana Fajri[1)]

[1)] *Program Studi Teknik Komputer Universitas Indo Global Mandiri*
Jl. Jend. Sudirman No. 629 KM.4 Palembang Kode Pos 30129
Email : rickymaulanafajri@uigm.ac.id[1)]

ABSTRACT

The use of the internet has boost up the interaction of the data and information. The internet has made the transfer of information become faster and reliable. However, there are many malicious applications which consider would violate the information. Therefore, there is a need for a network security to make sure the security on the internet. Network security has three main goals, namely confidentiality, integrity and availability. There are several techniques of an encryption, they are symmetric and asymmetric encryption. In symmetric encryption the key used to encrypt and decrypt a message is similar, while the key used in asymmetric encryption is different. There are many encryption types which are used in today life for example DES (Data encryption Standard), Triple DES and RSA. In this paper an application of encryption will be created. It will be created using PHP programming language and several stages of encryption for instance calculating the hash value using SHA-1, encrypt Hash value using RSA. Finally, the decryption process will be explored to make the encrypted message readable to the user.

Keyword: *Encryption, RSA, SHA-, Hash.*

1. Pendahuluan

Perkembangan teknologi komputer dan internet di Indonesia sangatlah pesat. Hal ini menunjukkan bahwa teknologi komputer dan internet memegang peranan penting dalam membantu pekerjaan manusia. komputer dan internet adalah teknologi yang telah menjadi bagian yang tidak dapat dipisahkan dari kehidupan manusia di era sekarang ini. Banyak pekerjaan yang menjadi lebih mudah, cepat dan akurat dengan bantuan teknologi tersebut.

Namun seiring dengan kemajuan teknologi internet banyak masalah yang ditemukan seperti virus komputer, *hacker*, *cracker* dan *spyware*. Hal tersebut sangat mengangu baik keamanan jaringan dan keamanan informasi dari sebuah perusahaan. Data dan informasi dari sebuah perusahaan adalah hal yang sangat penting, sehingga proses pertukaran data haruslah memperhatikan keamanannya. Keamanan pertukaran data meliputi keamanan media pertukaran dan keamanan data tersendiri.

Keamanan jaringan dapat dicapai melalui 3 prinsip yaitu *Confidentiality*, *Integrity*, dan *Availability*. *Confidentiality* adalah proses yang memastikan bahwa data tidak dapat dibaca oleh pihak lain selain pihak yang berkepentingan. *Confidentiality* dapat dilakukan dengan menggunakan metode enkripsi. Salah satu metode Enkripsi yang paling sering digunakan adalah metode RSA. RSA ditemukan pada tahun 1978 oleh Ron Rivest, Adi Shamir dan Len Adleman. The Rivest Shamir Adleman (RSA) adalah salah satu metode enkripsi yang cukup aman dalam melindungi keamanan data. Hal ini didasari dari metode RSA yang menggunakan pemfaktoran dua bilangan prima yang sangat besar. Sehingga sulit bagi peretas untuk memecahkan kode yang telah dienkripsi dengan metode RSA.

Berdasarkan penjelasan diatas, peneliti bermaksud untuk melakukan penelitian dengan judul rancang bangun aplikasi enkripsi dengan menggunakan metode RSA berbasis *web*. Diharapkan aplikasi yang dibangun dapat memudahkan pengguna dalam melakukan proses pertukaran data secara aman. Selain itu aplikasi yang dibangun dapat digunakan sebagai media informasi mengenai keamanan data dan jaringan khususnya ilmu kriptografi.

A. Perumusan Masalah

Berdasarkan latar belakang yang telah dikemukakan sebelumnya, maka dapat dirumuskan permasalahan yang menjadi objek penelitian ini diantaranya adalah : Bagaimana membangun sebuah aplikasi enkripsi dengan menggunakan metode RSA.

B. Batasan Masalah

Mengingat luasnya objek yang menjadi fokus penelitian ini, maka peneliti menentukan batasan permasalahan yang akan diteliti yaitu :

1. Metode enkripsi yang digunakan adalah RSA.
2. Data yang digunakan untuk dienkripsi bersifat teks.
3. Data yang digunakan untuk dienkripsi tidak lebih dari 8 byte.

C. Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah membangun sebuah aplikasi enkripsi dengan menggunakan metode RSA berbasis *web*.

ISSN PRINT : 2302-500X
ISSN ONLINE : 2477-3786

D.Manfaat Penelitian

Penelitian ini dapat memberikan manfaat diantaranya adalah

1. Dapat memperkaya ilmu keamanan jaringan khususnya kriptografi.
2. Dapat digunakan sebagai alternatif enkripsi data.

E. Tinjauan Pustaka

1) Pengertian Aplikasi

Aplikasi adalah perangkat lunak yang dibuat oleh suatu perusahaan computer untuk mengerjakan tugas-tugas tertentu misalnya *Ms.Word, Ms. Excel*. Aplikasi merupakan suatu program komputer yang berfungsi untuk menyelesaikan atau mengerjakan suatu pekerjaan atau permasalahan tertentu [1], sedangkan menurut Sungkono [2] *application software* (perangkat lunak aplikasi) terdiri atas program yang dirancang untuk membuat pengguna menjadi lebih produkti dan atau untuk membantu pengguna dengan tugas-tugas pribadi. Sehingga dapat disimpulkan bahwa aplikasi adalah suatu program komputer yang dibuat untuk memudahkan pekerjaan penggunanya.

2) Pengertian Keamanan Jaringan

Keamanan jaringan adalah melindungi jaringan , tetapi melindungi dalam hal ini adalah masih mempunyai artian luas. Keamanan tidak hanya tentang menjaga orang-orang di dalam jaringan dari dunia luar. Akan tetapi juga menyediakan akses ke dalam jaringan dengan cara yang dikehendaki, mempersilahkan orang-orang di dalam jaringan itu untuk bekerja sama.

3) Aspek-Aspek Keamanan Komputer

Ada beberapa aspek-aspek keamanan computer diantaranya adalah :

a. Authentication : agar penerima informasi dapat memastikan keaslian pesan tersebut dating dari orang yang dimintai informasi. Dengan kata lian informasi tersebut benar-benar dari orang yang dikehendaki.

b. Integrity : keaslian pesan yang dikirim melalui sebuah jaringan dan dapat dipastikan bahwa informasi yang dikirim tidak dimodifikasi oleh orang yang tidak berhak dalam perjalanan informasi tersebut.

c. Nonrepudiation : merupakan hal yang bersangkutan dengan si pengirim, si pengirim tidak dapat mengelak bahwa dialah yang mengirim informasi tersebut.

d. Authority : informasi yang berada pada sistem jaringan tidak dapat dimodifikasi oleh pihak yang todak berhak atas akses tersebut.

e. Confidentiality : merupakan usaha untuk menjaga informasi dari orang yang tidak berhak mengakses. *Confidentiality* biasanya berhubungan dengan informasi yang diberikan kepada pihak lain.

f. Privacy : merupakan lebih kearah data-data yang sifatnya privat (pribadi)

g. Availability : aspek availability atau ketersediaan berhubungan dengan ketersediaan informasi ketika dibutuhkan. sistem informasi yang diserang atau dijebol dapat menghambat atau meniadakan akses ke informasi.

h. Access control : aspek ini berhubungan dengan cara pengaturan akses kepada informasi. Hal itu biasanya berhubungan dengan masalah authentication dan juga privacy. Access control seringkali dilakukan menggunakan kombinasi user id dan password atau dengan menggunakan mekanisme lainnya.[3]

4) Metodologi Keamanan

Terdapat beberapa macam metodologi keamanan diantaranya adalah

a. Keamanan Level 0

Keamanan fisik merupakan keamanan tahap awal dari keamanan komputer. Jika keamanan fisik tidak terjaga dengan baik, maka data-data, bahkan perangkat keras komputer sendiri, tidak dapat diamankan.

b. Keamanan Level 1

Keamanan level 1 adalah keamanan data. *Data security* adalah cara mendesain *database* seaman mungkin. Seorang desainer *database* yang professional memikirkan kemungkinan-kemungkinan yang akan terjadi pada masalah keamanan dari database tersebut. Selanjutnya *device security* adalah alat-alat apa yang dipakai supaya keamanan dari komputer terjaga dan tidak kalah pentingnya adalah keamanan dari komputer itu sendiri.

c. Keamanan Level 2

Keamanan level 2 adalah *network security*. komputer yang terhubung dalam jaringan, baik itu LAN, WAN, maupun internet, sangat rawan dalam masalah keamanan karena komputer server bisa diakses menggunakan komputer client, baik itu merusak data, mencuri data, maupun melakukan perbuatan-perbuatan iseng lainnya.

d. Keamanan Level 3

Keamanan level 3 adalah *information security*. Maksud dari keamanan informasi di sini adalah keamanan informasi-informasi yang kadang kala tidak begitu diperdulikan oleh administrator atau pegawai, seperti memberikan password ke teman, kertas-kertas bekas transaksi dan lain sebagainya. Hal tersebut bisa menjadi sesuatu yang sangat fatal. Jika informasi-informasi tersebut diketahui oleh orang-orang yang tidak bertanggung jawab, maka mereka akan dapat mengakses data-data penting yang ada di dalam computer.

e. Keamanan Level 4

Keamanan level 4 merupakan keamanan secara keseluruhan dari computer. Jika level 1-3 sudah dapat dikerjakan dengan baik, maka otomatis keamanan untuk level 4 sudah terpenuhi. Akan tetapi jika salah satu dari level tersebut belum bisa terpenuhi, maka masih ada lubang keamanan yang bisa diakses.[3]

5) Kriptografi

a. Sejarah kriptografi

Kriptografi berasal dari bahasa yunani, crypto dan graphia. Crypto berarti secret (rahasia) dan graphia berarti writing (tulisan). Menurut terminologynya kriptografi adalah ilmu dan seni untuk menjaga keamanan pesan ketika pesan dikirim dari suatu

ISSN PRINT : 2302-500X
ISSN ONLINE : 2477-3786

tempat ke tempat lain. Kriptografi sudah digunakan 4000 tahun yang lalu, diperkenalkan oleh orang-orang mesi lewat hieroglyph. [4]

b. Algoritma Kriptografi

Algoritma kriptografi terdiri dari tiga fungsi dasar yaitu :

Enkripsi : merupakan pengaman data yang dikirimkan agar terjaga kerahasiaannya. Pesan asli disebut *plain text*, yang diubah menjadi kode-kode yang tidak dimengerti. Enkripsi bisa diartikan dengan *cipher* atau kode.

Deskripsi : merupakan kebalikan dari enrkipsi. Pesan yang telah dienkripsi dikembalikan ke bentuk asalnya (teks asli), disebut dengan dekripsi pesan. Algoritma yang digunakan untuk dekripsi tentu berbeda dengan algoritma yang digunakan untuk enkripsi.

Kunci : kunci adalah sesuatu yang digunakan untuk membuat enkripsi atau dekripsi pesan. Kunci yang digunakan terbagi menjadi dua bagian kunci rahasia (private key) dan kunci umum (public key) [4]

c. Jenis-jenis algoritma kriptografi

Algoritma kriptografi dibagi menjadi 3 bagian berdasarkan kunci yang dipakai.

Algoritma Simetri

Algoritma ini sering disebut dengan algoritma klasik karena memakai kunci yang sama untuk enkripsi dan dekripsi. Bila mengirim pesan dengan menggunakan algoritma ini, si penerima pesan harus mendapatkan informasi mengenai kunci dari pesan agar dapat mendekripsi pesan yang dikirim. Keamanan dari pesan yang menggunakan algoritma ini tergantung dari kunci. Jika kunci tersebut diketahui oleh orang lain, maka orang tersebut dapat melakukan enkripsi dan dekripsi terhadap pesan. Jenis enkripsi yang menggunakan algoritma simetri adalah : *Data Encryption Standard (DES), RC2, RC4,RC5,RC6, International Data Encryption Algorithm (IDEA), Advanced Encryption Standard (AES), One Time Pad (OTP).*

Algoritma Asimetri

Algoritma asimetri sering juga disebut dengan *public key encryption*, dengan arti kata kunci yang digunakan untuk melakukan enkripsi dan dekripsi berbeda. Pada algoritma asimetri kunci dibagi menjadi dua bagian. Kunci Umum (*Public Key*) dan Kunci Rahasia (*Private Key*). Kunci-kunci tersebut berhubungan satu sama lain. Dengan kunci *public* orang dapat mengenkripsi pesan tetapi tidak bisa mendekripsikannya. Hanya orang yang memiliki kunci rahasia yang dapat mendekripsikan pesan tersebut. Jenis enkripsi yang menggunakan algoritma asimetri adalah : *Digital Signature Algorithm (DSA), RSA, DIffie-Hellman(DH), Elliptic Curve Cryptography (ECC), Kriptografi Quantum*.

Hash Function

Fungsi hash merupakan suatu fungsi matematika yang mengambil masukkan panjang variabel dan mengubahnya ke dalam urutan biner dengan panjang yang tetap. Fungsi hash biasanya diperlukan bila ingin membuat sidik jari dari suatu pesan. Sidik jari pada pesan merupakan suatu tanda bahwa pesan tersebut benar-benar berasal dari orang yang diinginkan.

6) Web

World wide web atau *web* merupakan sumber daya internet yang sangat popular dan dapat digunakan untuk memperoleh informasi atau bahkan melakukan transaksi pembelian barang.

Web menggunakan protocol yang disebut HTTP (*Hyper Text Transfer Protocol*) yang berjalan pada TCP/IP. Adapun dokumen *web* ditulis dalam format HTML (Hypertext Markup Language). Dokumen ini diletakkan dalam *web Server (Server* yang melayani permintaan halam *Web*) dan Diakses oleh klien (pengakses informasi) melalui perangkat lunak yang disebut *web* browser. [5]

2. Pembahasan

A. Metodologi Penelitian

1) Alat yang digunakan

Adapun peralatan yang digunakan dalam penelitian ini adalah : Komputer dengan spesifikasi minimal intel Pentium IV, Printer, Perangkat Lunak yang digunakan adalah WAMPP (*web* server), Netbeans 6.9.1, GIMPP

2) Jalan Penelitian

Penelitan ini dilakukan dalam beberapa tahap yaitu

1. Menganalisa kebutuhan aplikasi.
2. Menggumpulkan data yang dibutuhkan seperti fungsi hash, public key, private key.
3. Membuat desain sistem aplikasi.
4. Implementasi aplikasi.
5. Pengujian aplikasi.
6. Evaluasi.

3) Metode Pengembangan Aplikasi

Dalam Penelitian ini metode pengembangan sistem yang digunakan adalah model *waterfall*. tahap-tahap dari model *waterfall* adalah sebagai berikut :

1. Analisis dan definisi Persyaratan
   Pada tahap ini fungsi, batasan dan tujuan sistem ditentukan berdasarkan analisa kebutuhan yang dikemukakan oleh pengguna sistem.
2. Perancangan sistem dan perangkat lunak
   Pada tahap ini sistem mulai dirancang meliputi sistem perangkat keras atau pun lunak. Pada tahap ini arsitektur sistem didesain secara keseluruhan.
3. Implementasi dan pengujian unit
   Pada tahap ini, perancangan perangkat lunak mulai diimplemtasikan kedalam bentuk kode program . Pengujian dilakukan terhadap unit-unit program yang berfungsi memastikan unit program tersebut berjalan dengan sebagaimana mestinya.
4. Integrasi dan pengujian sistem
   Pengujian sistem secara keseluruhan dilakukan pada tahap ini, pengujian bertujuan sistem dapat berjalan secara keseluruhan. Setelah melewati tahap ini, sistem dapat dikirim kepada pengguna.

ISSN PRINT : 2302-500X
ISSN ONLINE : 2477-3786

5. Operasi dan pemeliharaan
   Tahap ini adalah tahap terakhir dalam model waterfall. Pada tahap ini sistem telah dipakai oleh user, pemeliharaan sistem dilakukan dengan cara mendengar keluhan dari pemakai. Sehingga error yang ditemukan dapat diperbaiki.[6]

B. Analisis Kebutuhan

Pada tahap ini kebutuhan terhadap sistem haruslah di analisis. Analisis meliputi analisis kebutuhan perangkat lunak, analisis kebutuhan perangkat keras, dan analisis kebutuhan aplikasi.

1) Analisis kebutuhan perangkat lunak

Pada tahap ini ditentukan kebutuhan perangkat lunak yang akan digunakan dalam membangun sebuah aplikasi enkripsi. Adapun perangkat lunak yang diperlukan dalam membangun aplikasi enkripsi berbasis RSA adalah sebagai berikut :

- *Web* server apache (xampp)
- Code editor (netbean)
- *Web* browser (Mozilla firefox)
- Sistem operasi (Microsoft windows xp professional sp3)

2) Analisis kebutuhan perangkat keras

Setelah kebutuhan perangkat lunak selesai dianalisis, maka tahap selanjutnya adalah menganalisis kebutuhan perangkat keras yang akan digunakan. Perangkat keras yang digunakan harus mampu mendukung semua perangkat lunak yang digunakan akan pembangunan aplikasi dapat berjalan dengan baik. Adapun perangkat keras yang digunakan adalah sebagai berikut :

- Perangkat komputer dengan spesifikasi processor intel Pentium 4
- Hard Disk minimal 40 GB
- Memory 1 GB
- Monitor
- Keyboard dan Mouse

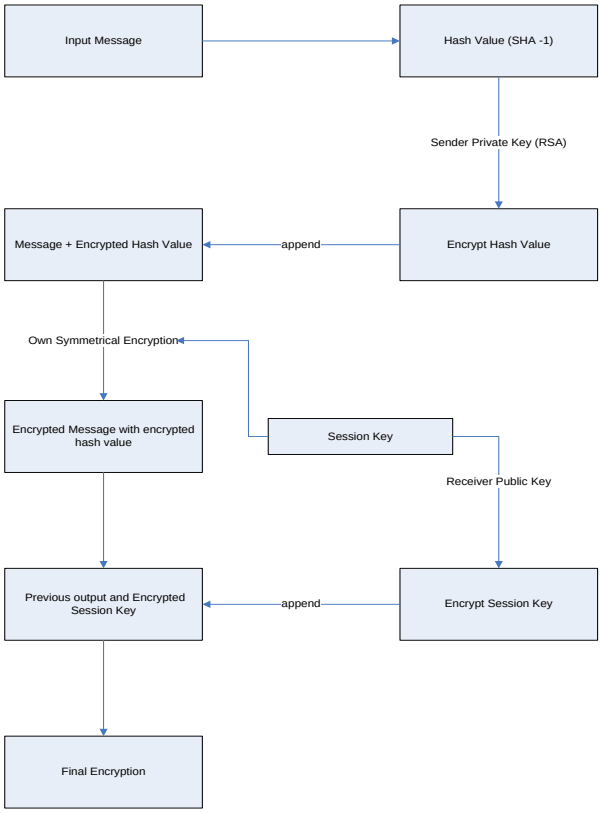
3) Analisis Kebutuhan Aplikasi

Analisis terhadap kebutuhan aplikasi adalah sebuah hal wajib yang harus dikerjakan agar aplikasi yang dibangun dapat digunakan dengan sebagaimana mestinya. Pada tahap ini ditentukan spesifikasi minimal dari aplikasi. kebutuhan minimal dari aplikasi enkripsi berbasis RSA adalah sebagai berikut

- Aplikasi dapat melakukan proses enkripsi data.
- Aplikasi dapat melakukan proses dekripsi data.
- Aplikasi dapat mengembalikan data yang dienkripsi melalui proses dekripsi.

4) Desain Sistem

Sebelum sebuah sistem dapat diimplementasikan kedalam bentuk kode program, sebuah aplikasi harus lah didesain terlebih dahulu. Proses desain adalah proses menterjemahkan kebutuhan-kebutuhan aplikasi kedalam bentuk simbol-simbol konseptual. Simbol-simbol tersebut akan diterjemahkan kedalam bentuk kode program, sehingga kode program akan sesuai dengan kebutuhan dari sistem informasi. Pada tahap ini, peneliti menggunakan *flow* chart atau diagram alir. Diagram alir mengambarkan proses-proses yang dilakukan oleh aplikasi enkripsi berbasis RSA.



Gambar 1. *FlowChart Enkripsi*

C. Proses Enkripsi

1) Hash Value

Proses pertama dalam aplikasi enkripsi ini adalah menghitung nilai hash (*hash value*) dari pesan yang dimasukan. Pada aplikasi ini, metode penghitungan hash value yang dipilih adalah *SHA-1(Secured Hash Algorithm*). *SHA-1* adalah metode penghitungan nilai has yang banyak digunakan oleh beberapa protocol security dan aplikasi. SHA dibuat pada tahun 1993 oleh the National Institute of Standard and Technology ini FIPS-180.

2) Enkrip Nilai HASH dengan private key yang dimiliki oleh pengirim

Setelah menghitung nilai hash, proses selanjutnya adalah melakukan enkripsi terhadap hash value tersebut. Pada proses ini, hash value akan dienkripsi dengan menggunakan kunci privat yang dimiliki oleh pengirim. Proses enkripsi ini didasari dari enkripsi asimetri. Pada proses ini akan menggunakan metode RSA untuk melakukan enkripsi.

3) Mengabungkan Hash Value dan Pesan

Selanjutnya adalah penggabungan hash value yang telah dienkripsi ke pesan original, sehingga terdapat dua

ISSN PRINT : 2302-500X
ISSN ONLINE : 2477-3786

pesan, yaitu pesan original dan hash value yang telah dienkripsi.

4) Enkripsi pesan original dengan session key

Selanjutnya pesan original dan hash value yang telah dienkripsi akan dienkripsi lagi menggunakan sebuah session key yang diacak. Session key tersebut akan dibuat menggunakan code php menggunakan input acak yang berasal dari 32 bit String acak. Pada proses ini, penulis menggunakan metode enkripsi sebagai berikut :

1. *Subsitusi byte*
   Pada tahap ini, setiap byte dari 128 (16 byte) akan diganti dengan byte lain yang diambil dari s-box. Sehingga proses ini memeberikan sebuah chipper text yang sulit dikembalikan.
2. *Shift row*
   Pada tahap ini, setiap byte pada baris nya akan diacak lagi. Pada baris pertama byte tidak akan diubah, sehingga perubahan baris akan dilakukan pada baris kedua, baris ketiga dan selanjutnya.
3. *Mix Columns*
   Pada tahap ini, setiap kolom akan diacak lagi dan di simpan, sehingga menimbulkan sebuah chipper text yang sulit untuk didekripsi.
4. *Add Round Key*
   Terakhir sebuah subkey akan degenerate sehingga akan menambah tingkat kesulitan sebuah pesan untuk dijebol.

5) Electrical Code Book (ECB)

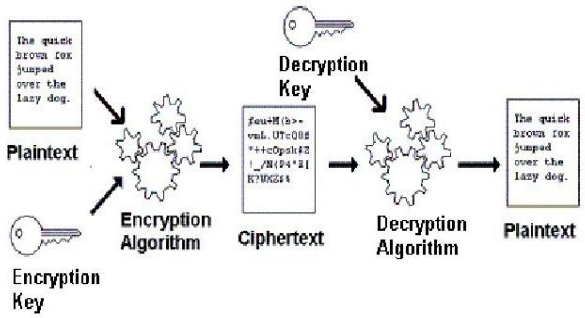
Selanjutny pesan dapat dienkripsi menggunakan session key. ECB adalah sebuah metode yang enkripsi yang sangat baik. ECB adalah sebuah metode enkripsi yang cepat. Namun pada proses enkripsi sebuah blok pesan akan dienkripsi menggunakan sebuah key yang sama. Sehingga hal ini akan mengasilkan sebuah plaintext yang sama dengan *chipper text*. Sehingga metode ini sangat rentan untuk dijebol menggunakan replay attacks.

6) Chain Block Cipher (CBC)

Enkripsi CBC juga menggunakan session key untuk menenkrip pesan. Namun sebelum pesan dienkripsi, akan dilakukan proses *XOR* dengan sebuah nilai acak, selanjutnya blok kedua dari pesan akan juga dilakukan proses *XOR* berdasarkan blok sebelumnya. Sehingga setiap blok akan menghasilkan sebuah pesan yang berbeda. Jadi jika pesan diubah sedikit, maka seluruh blok dari *CBC* akan juga ikut berubah.

7) Enkripsi Session key

Setelah pesan asli dan hash value selesai dienkripsi, proses selanjutnya adalah mengenkripsi session key. Berdasarkan penjelasan sebelumnya session key adalah sebuah nilai random yang dibuat menggunakan *PHP code*. Pada enkripsi asimetri, sebuah key yang berbeda yang akan digunakan oleh pengirim dan penerima pesan. Sehingga pada enkripsi asimetri key penerima pesan harus disimpan secara rahasia. Sehingga kerahasiaan pesan dapat dijaga.

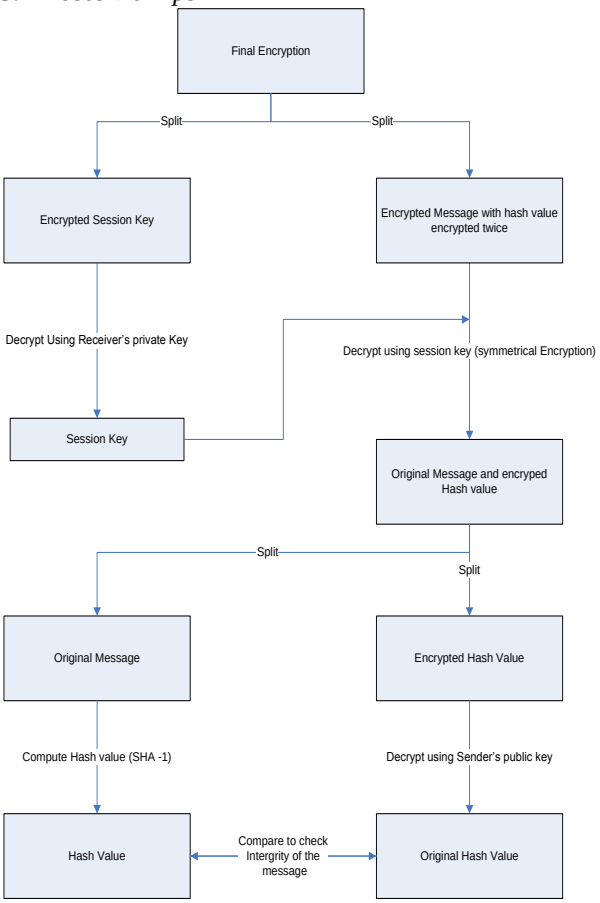


Gambar 2. *Algoritma Asimetri*

8) Output

Proses ini adalah proses terakhir dari proses enkripsi pesan. Selanjutnya hasil output akan dikirim ke penerima pesan.

C. Proses Dekripsi



Gambar 3. *Proses Dekripsi*

1) Pisahkan session key dari chipper text

Proses dekripsi pesan dimulai dengan memisahkan *session key* yang telah dienkripsi. *Session key* adalah sebuah *key* yang di hasilkan dari sebuah *PHP code* acak. *Session key* ini ditambahkan ke chipper text.

2) Dekripsi session key dengan private key penerima

Session key sebelumnya yang telah dienkripsi menggunakan *public key* dari penerima menggunakan metode RSA. Setelah memisakhan *session key* dari pesan original, sekarang penerima dapat mendekripsi session

ISSN PRINT : 2302-500X
ISSN ONLINE : 2477-3786

key dengan *private key* dari penerima. Sehingga pada proses ini session key telah berhasil didekripsi, sehingga *session key* yang asli telah didapatkan.

3) Dekripsi pesan dengan menggunakan session key

Setelah *session key* berhasil didekripsi, selanjutnya pesan yang telah enkripsi dapat di dekripsi menggunakan teknik dekripsi sendiri untuk mendapatkan pesan asli dan hash value yang telah dienkripsi. Selanjutnya aplikasi harus memisahkan hash value dengan pesan original

4) Dekripsi hash value

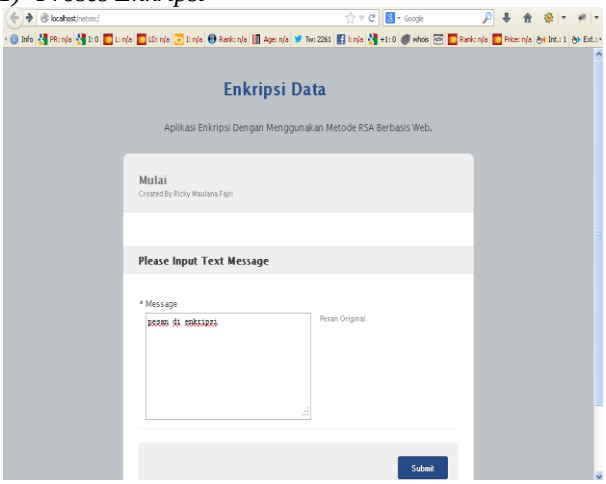
Setelah memisahkan *hash value* dengan pesan original, proses selanjutnya adalah mendekripsi hash value menggunakan public key dari pengirim pesan, setelah proses ini aplikasi akan mendapakan hash value asli dan pesan original

5) Cek integritas dari pesan.

Setelah mendapatkan *hash value* original dari pesan yang telah dienkripsi, proses selanjutnya adalah untuk melihat apakah pesan tidak mengalami perubahan. Untuk melihat apakah pesan tidak mengalami perubahan perlu dilakukan proses penghitungan hash value, jika hash value yang dihasilkan sama maka dapat dikatakan bahwa pesan tidak mengalami perubahan.

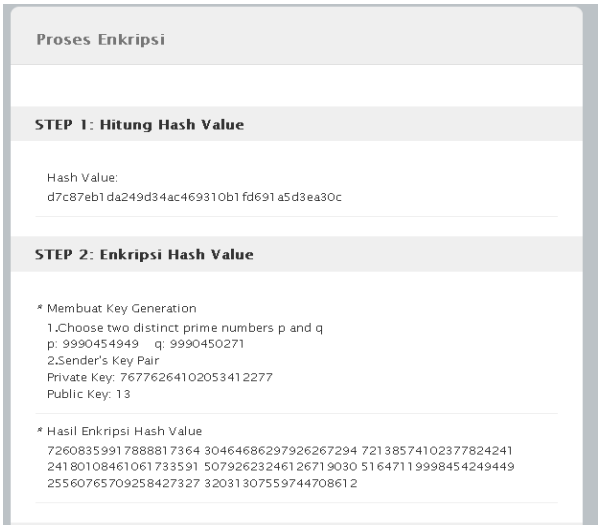
D. Hasil dan Pembahasan

1) Proses Enkripsi



Gambar 4. *Tampilan Utama Web*

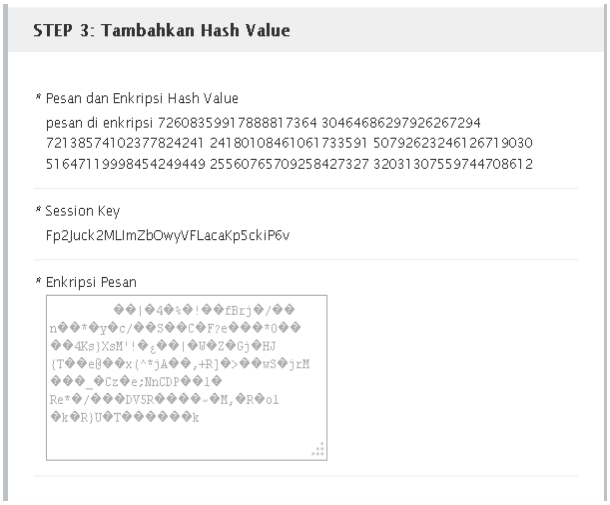
Pada tampilan utama terdapat sebuah pesan yang akan dienkripsi. Pada contoh diatas penulis memasukkan teks "pesan di enkripsi" proses selanjutnya adalah menekan tombol submit sehingga proses enkripsi akan dilakukan



Gambar 5. *Tampilan Proses Enkripsi*

Pada step 1 adalah proses menghitung *hash value*. Sesuai dengan flowchart yang digambarkan pada bagian sebelumnya. Proses menghitung *hash value* adalah proses penghitungan sebuah nilai untuk mengecek apakah sebuah pesan telah diubah oleh orang lain.

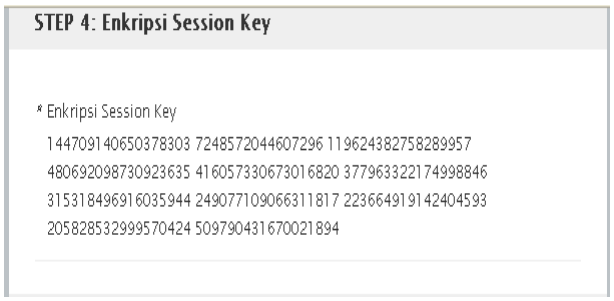
Pada step dua adalah proses mengenkripsi hash value. Pada proses ini terbagi menjadi dua tahapan 1. Adalah memilih sebuah bilangan prima terbesar (bilangan tersebut diacak) selanjutnya mencari nilai kunci ganda yaitu *private key* dan *public key*. Selanjutnya *hash value* yang dihasilkan dari step pertama akan dienkripsi menggunakan key ganda tersebut.



Gambar 6. *Step 3 Tambahkan Hash Value*

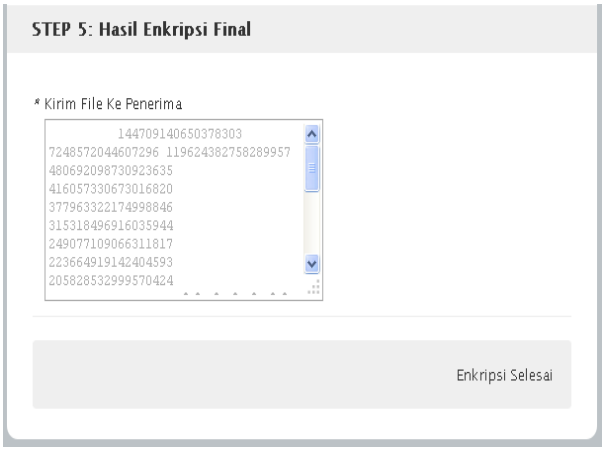
Proses pada step 3 adalah menambahkan *hash value* yang telah dienkripsi kedalam pesan, sehingga nilai yang didapatkan adalah sebuah nilai acak yang sulit dibaca. Pada step 3 ini, aplikasi akan menghasilkan sebuah *session key*.

ISSN PRINT : 2302-500X
ISSN ONLINE : 2477-3786



Gambar 7. *Hasil Enkripsi session key*

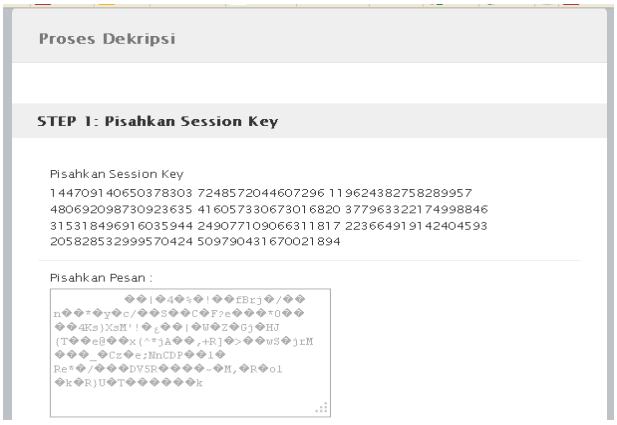
Pada step 3 telah dijelaskan bahwa aplikasi akan mengenerate sebuah session key. *Session key* ini berasal dari pemrograman php. Selanjutnya *session key* yang dihasilkan akan dienkripsi lagi. Enkripsi session key akan ditambahkan kepada pesan akhir yang akan dikirim ke penerima.



Gambar 8. *Hasil Enkripsi*

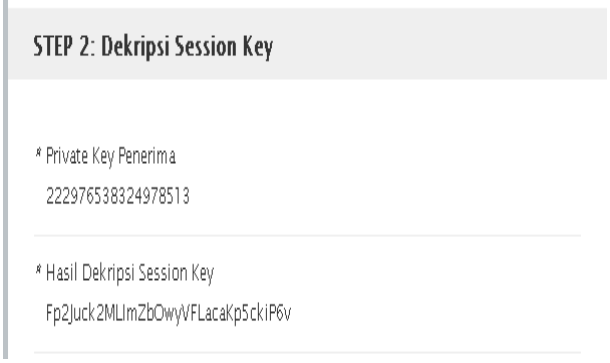
Step 5 adalah hasil akhir dari step-step sebelumnya sehingga hasil dari step 5 ini yang akan dikirim ke penerima. Selanjutnya penerima akan mendekripsi pesan sehingga didapatkan pesan yang sama sebelum di enkripsi.

2) Proses Dekripsi



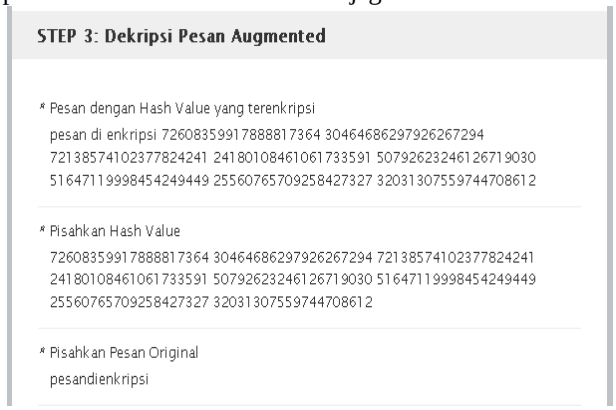
Gambar 9. *Step 1 Dekripsi*

Pada step 1 proses dekripsi adalah pemisahan *session key* dengan pesan. Proses pemisahan ini tidak dapat dilakukan oleh aplikasi lain jika aplikasi lain tersebut tidak memiliki sebuah algoritma enkripsi dan dekripsi yang sama. Pada proses ini session key dan pesan akan dipisahkan sehingga aplikasi akan mendapatkan dua data baru yaitu session key dan pesan yang telah dienkripsi.



Gambar 10. *Proses Dekripsi Session Key*

Pada step kedua aplikasi akan mendekripsi session key, sehingga aplikasi akan mendapatkan sebuah session key yang sesuai dengan pada saat session key yang belum di enkripsi. Proses dekripsi ini menggunakan *private key* dari penerima. Sehingga kerahasiaan dari *private key* penerima haruslah senantiasa terjaga.



Gambar 11. *Dekripsi Pesan*

Pada step ketiga proses pemisahan pesan original dan hash value. Hasil akhir dari proses ini adalah didapatkan sebuah pesan asli yang telah dienkripsi namun, proses belum selesai. Meskipun pesan asli telah didapatkan, proses terakhir adalah melakukan pengecekan hash value. Proses pengecekan hash value bertujuan untuk memastikan bahwa pesan tidak diubah.

Gambar 12. *Dekripsi Hash Value*

Pada penjelasan pada step 4, aplikasi akan menggunakan *hash value* untuk melakukan pengecekan terhadap pesan. Apakah pesan telah diubah atau tidak. Namun pada aplikasi ini *hash value* telah dienkrip oleh aplikasi.

ISSN PRINT : 2302-500X
ISSN ONLINE : 2477-3786

Sehingga aplikasi perlu melakukan proses dekripsi untuk mendapatkan nilai hash value yang asli. Step 4 adalah proses untuk mendekripsi hash value sehingga didapatkan nilai hash value yang dapat digunakan untuk melakukan pengecekan terhadap pesan.

Gambar 13. *Check Integritas dari Pesan*

Pada step 5 adalah memastikan bahwa pesan tidak diubah melalui penghitungan hash value. Pada gambar diatas dapat dicek bahwa pesan tidak mengalami perubahan. (Penghilangan spasi dilakukan untuk mempercepat perhitungan hash value). Sehingga dapat dikatakan proses enkripsi dan dekripsi selesai dilakukan.

D. Pengujian Aplikasi

Pengujian aplikasi yang dilakukan dalam penelitian ini menggunakan metode black box. Metode black box menguji semua menu yang ada didalam aplikasi, apakah menu tersebut berfungsi atau tidak.

Tabel 1. *Pengujian Aplikasi dengan BlackBox*

No	Menu Yang Diuji	Status
1	Menu Halaman Utama	Berhasil
2	Menu submit enkripsi	Berhasil
3	Enkripsi data 8 string character	Berhasil
4	Enkripsi data 16 string character	Berhasil
5	Enkripsi data 32 string character	Berhasil
6	Enkripsi data 64 string character	Berhasil
7	Enkripsi data 128 string character	Berhasil
8	Enkripsi data 256 string character	Berhasil
9	Enkripsi data 512 string character	Berhasil
10	Enkripsi data 1024 string character	Berhasil
11	Enkripsi data 2048 string character	Berhasil
12	Enkripsi data 3000 string character	Berhasil
13	Enkripsi data 4000 string character	Berhasil
14	Enkripsi data 5000 string character	Berhasil
15	Enkripsi data 6000 string character	Berhasil
16	Enkripsi data 7000 string character	Berhasil

3. Kesimpulan

A. Kesimpulan

1. RSA adalah sebuah metode enkripsi yang ditemukan oleh Ron Rivest, Adi Shamir dan Len Adleman pada tahun 1978
2. RSA adalah sebuah metode algoritma enkripsi asimetri
3. RSA adalah sebuah metode algoritma enkripsi data yang sangat baik untuk digunakan untuk menjaga kerahasiaan data.
4. RSA termasuk algoritma yang cepat efektif dan efisien

B. Saran

1. Tampilan aplikasi perlu ditambahkan lagi sehingga fungsi dapat diperbanyak.
2. Perlu ditambahkan form lagi sehingga input text yang akan dienkripsi menjadi lebih banyak.
3. Perlu ditambahkan algoritma lain sehingga dapat meningkatkan tingkat kerahasiaan data.

Daftar Pustaka

[1] Dhanta, Rizky, 2009 "*Kamus Istilah Komputer Grafis & Internet*" Surabaya Indah.

[2] Sungkono, Chriswan (Penerjemah), 2007 " *Menjelajah Dunia Komputer Fundamental Edisi 3*" Jakarta, Salemba Infotek.

[3] Ariyus, Dony, 2008 " *Pengenalan Ilmu Kriptografi Teori Analisis dan Implementasi*", Andi, ISBN 978-979-29-0477-2

[4] Kadir, Abdul & Triwahyuni, Terra CH 2003-2005 " *Pengenalan Teknologi Informasi*" Andi Jogya, ISBN 979-731-637-8

[5] Sommervile, Ian 2001, "*Software Engineering, 6th edition*", Addison Wesly, ISBN 0-201-39815-X